Meet Magento

eCommerce Platform for Growth

meet-magento.nl

#MM15NL

Gordon Lesti
Recommender Systems

- Student
- @GordonLesti
- gordonlesti.com

Typical problems

- What other items do customer buy after viewing this item
- Who to follow
- People who viewed this item also viewed
- Related or similar movies?
- ...

Basic concepts

- Collaborative recommendation
- Content-based recommendation
- Knowledge-based recommendation
- Hybrid recommendation

Compare items

$$Item1 \Rightarrow \begin{pmatrix} 1 \\ 2 \\ 5 \\ 3 \end{pmatrix},\ Item2 \Rightarrow \begin{pmatrix} 4 \\ 3 \\ 4 \\ 1 \end{pmatrix},\ Item3 \Rightarrow \begin{pmatrix} 2 \\ 1 \\ 4 \\ 5 \end{pmatrix}$$

Cosine similarity measure

$$sim(\overrightarrow{a}, \overrightarrow{b}) = \frac{\overrightarrow{a} \cdot \overrightarrow{b}}{|\overrightarrow{a}| * |\overrightarrow{b}|}$$

- $sim(Item1, Item2) = \frac{1*4+2*3+5*4+3*1}{\sqrt{1^2+2^2+5^2+3^2}*\sqrt{4^2+3^2+4^2+1^2}} \approx 0.8153742483272117$
- $sim(Item1, Item3) = \frac{1*2+2*1+5*4+3*5}{\sqrt{1^2+2^2+5^2+3^2}*\sqrt{2^2+1^2+4^2+5^2}} \approx 0.9207747210672769$
- $sim(Item1, Item3) = \frac{4*2+3*1+4*4+1*5}{\sqrt{4^2+3^2+4^2+1^2}*\sqrt{2^2+1^2+4^2+5^2}} \approx 0.7280252083092640$

	Item1	*Item2*	*Item3*
Item1	1.0	0.815	0.921
Item2	0.815	1.0	0.728
Item3	0.921	0.728	1.0

Conventions

- $U = \{u_1, \ldots, u_n\}$ set of users
- $P = \{p_1, \ldots, p_m\}$ set of products (items)
- R is a $n \times m$ matrix of ratings $r_{i,j}$ with $i \in 1 \ldots n$, $j \in 1 \ldots m$

Collaborative recommendation

If users shared the same interests in the past - [. . .] - they will also have similar tastes in the future (Jannach et al. 2010)

Example

- Build $R = U \times I$

$$R = \begin{pmatrix} 3 & 0 & 0 & 2 & 5 & 0 \\ 2 & 0 & 0 & 1 & 0 & 0 \\ 0 & 4 & 1 & 3 & 0 & 5 \\ 0 & 1 & 4 & 0 & 5 & 2 \end{pmatrix}$$

- Calculate SVD from R

$$U = \begin{pmatrix} 0.50044973 & 0.53419723 & 0.58856666 & -0.34318023 \\ 0.07586901 & 0.05106193 & 0.41654165 & 0.90450518 \\ 0.5423152 & -0.80659153 & 0.21362373 & -0.0983322 \\ 0.67058794 & 0.24786305 & -0.65912612 & 0.23329914 \end{pmatrix}$$

$$S = \begin{pmatrix} 8.90022547 & 0 & 0 & 0 \\ 0 & 6.42689669 & 0 & 0 \\ 0 & 0 & 4.22402935 & 0 \\ 0 & 0 & 0 & 1.28006315 \end{pmatrix}$$

$$V = \begin{pmatrix} 0.18573543 & 0.31907605 & 0.36231295 & 0.3037804 & 0.65786966 & 0.45535384 \\ 0.26524708 & -0.46344344 & 0.0287636 & -0.20232443 & 0.60842761 & -0.5503794 \\ 0.61523798 & 0.04625176 & -0.57359467 & 0.5290082 & -0.0835215 & -0.05921682 \\ 0.60893065 & -0.125017 & 0.65220559 & -0.06003755 & -0.42920184 & -0.01957928 \\ -0.37305791 & -0.44764731 & 0.27772459 & 0.74611583 & -0.07461158 & -0.14509657 \\ 0.08160772 & -0.68209535 & -0.19075651 & -0.16321543 & 0.01632154 & 0.68175684 \end{pmatrix}$$

- Reduce rank to k (3 in this example)

$$U_k = \begin{pmatrix} 0.50044973 & 0.53419723 & 0.58856666 \\ 0.07586901 & 0.05106193 & 0.41654165 \\ 0.5423152 & -0.80659153 & 0.21362373 \\ 0.67058794 & 0.24786305 & -0.65912612 \end{pmatrix}$$

$$S_k = \begin{pmatrix} 8.90022547 & 0 & 0 \\ 0 & 6.42689669 & 0 \\ 0 & 0 & 4.22402935 \end{pmatrix}$$

$$V_k = \begin{pmatrix} 0.18573543 & 0.31907605 & 0.36231295 \\ 0.26524708 & -0.46344344 & 0.0287636 \\ 0.61523798 & 0.04625176 & -0.57359467 \\ 0.60893065 & -0.125017 & 0.65220559 \\ -0.37305791 & -0.44764731 & 0.27772459 \\ 0.08160772 & -0.68209535 & -0.19075651 \end{pmatrix}$$

- Calculate $A = U_k \cdot S_k^{\frac{1}{2}}$
- row i contains user i

$$\begin{pmatrix} 1.49300397 & 1.35426076 & 1.20964831 \\ 0.22634187 & 0.12944875 & 0.85609488 \\ 1.61790226 & -2.0448164 & 0.43904897 \\ 2.00058148 & 0.62836567 & -1.35466523 \end{pmatrix}$$

- Calculate WU with $WU_{i,j} = sim(A_i, A_j)$

$$WU = \begin{pmatrix} 1 & 0.736197598021989 03 & 0.0285418079268030 67 & 0.374729825035962 31 \\ 0.73619759802198903 & 1 & 0.20173345700472975 & -0.28000362621299618 \\ 0.028541807926803067 & 0.20173345700472975 & 1 & 0.20558656126701419 \\ 0.37472982503596231 & -0.28000362621299618 & 0.20558656126701419 & 1 \end{pmatrix}$$

- Calculate $WU \cdot R$

$$\begin{pmatrix} 4.4723952 & 0.48889706 & 1.52746111 & 2.82182302 & 6.87364913 & 0.89216869 \\ 4.20859279 & 0.5269302 & -0.91828105 & 3.07759557 & 2.28096986 & 0.44866003 \\ 0.48909234 & 4.20558656 & 1.82234625 & 3.25881707 & 1.17064185 & 5.41117312 \\ 0.56418222 & 1.82234625 & 4.20558656 & 1.08621571 & 6.87364913 & 3.02793281 \end{pmatrix}$$

- Remove already rated entries
- Choose the product j with the highest value for user i

$$
\begin{pmatrix}
 & 0.48889706 & 1.52746111 & & & 0.89216869 \\
 & 0.5269302 & -0.91828105 & & 2.28096986 & 0.44866003 \\
0.48909234 & & & & 1.17064185 & \\
0.56418222 & & & 1.08621571 & &
\end{pmatrix}
$$

Advantages

- recommendations between categories
- classification by users

Disadvantages

- cold-start problem
  - new-user problem
  - new-item problem
  - filling up empty matrices
- Lemming effect
- performance

Other weighting metrics

- Pearson's correlation coefficient: $sim(a,b) = \frac{\sum_{p \in P}(r_{a,p} - \overline{r_a})(r_{b,p} - \overline{r_b})}{\sqrt{\sum_{p \in P}(r_{a,p} - \overline{r_a})^2}\sqrt{\sum_{p \in P}(r_{b,p} - \overline{r_b})^2}}$
- Adjusted cosine similarity: $sim(a,b) = \frac{\sum_{u \in U}(r_{u,a} - \overline{r_u})(r_{u,b} - \overline{r_u})}{\sqrt{\sum_{u \in U}(r_{u,a} - \overline{r_u})^2}\sqrt{\sum_{u \in U}(r_{u,b} - \overline{r_u})^2}}$
- ...

Content-based recommendation

Find items that are most similar to my current item absed on their content

Example Latent Semantic Indexing (LSI)

- Text1 *How to create a filter in magento by category products*
- Text2 *Filter product collection by multiple categories*
- Text3 *How to filter products NOT IN categories*
- Text4 *Filter category page products on Sidebar*
- Text5 *Filter Child Categories by Their Products Attribute*

- split text into words
- remove special character, filter (all lowercase)
- replace synonyms
- weight important words for example with TF-IDF
- ...

	Text1	Text2	Text3	Text4	Text5
how	1	0	1	0	0
to	1	0	1	0	0
create	1	0	0	0	0
a	1	0	0	0	0
filter	1	1	1	1	1
in	1	0	1	0	0
magento	1	0	0	0	0
by	1	1	0	0	1
category	1	0	0	1	0
products	1	0	1	1	1
product	0	1	0	0	0
collection	0	1	0	0	0
multiple	0	1	0	0	0
categories	0	1	1	0	1
not	0	0	1	0	0
page	0	0	0	1	0
on	0	0	0	1	0
child	0	0	0	0	1
their	0	0	0	0	1
attribute	0	0	0	0	1

- compare texts with a metric

	Text1	Text2	Text3	Text4	Text5
Text1	1	0.258	0.597	0.424	0.358
Text2	0.258	1	0.308	0.182	0.462
Text3	0.597	0.308	1	0.338	0.428
Text4	0.424	0.182	0.338	1	0.338
Text5	0.358	0.462	0.428	0.338	1

Advantages

- no cold-start problem
- simple implementation

Disadvantages

- items have to be filled with content
- difficult for complexe items

Knowledge-based recommendation

- Filter
- Customer attributes
- . . .

Hybrid recommendation

- mix different concepts

Implementation

- Numpy (numpy.org) [fast]
- GNU Octave (www.gnu.org/s/octave/) [fast]
- NumPHP (numphp.org) [slow]

References

- Dietmar Jannach, Markus Zanker, Alexander Felfernig & Gerhard Friedrich, 2010, *Recommender Systems: An Introduction*, Cambridge University Press, New York

Questions?

Thanks for listening